Design Tokens Overview

Design Tokens Play+ Design Tokens: The Complete Guide ■ Introduction ■ Design tokens are the atoms of the Play+ design system. This guide follows a tiered token architecture , enabling scalable, consistent, and fully themeable UI components across experiences. Play+ embraces a CSS-first model for theming. Users define branding through a simple mapping of semantic tokens to raw values. Everything else—such as hover states, dark mode variants, and accessibility contrast—is derived automatically by Play+. Token File Locations ■ styles/tokens/ : Contains all Play+ global design tokens (color, typography, spacing, etc.) styles/tokens/themes/ : Contains per-brand theme overrides styles/tokens/components/ : Contains component-specific semantic tokens Note: Never modify core Play+ token files directly. Only customize branding by editing your theme files. Why Use Tokens? ■ Tokens are named values used instead of hardcoded literals. They: Ensure design consistency across the UI Enable runtime theming and brand customization Power Play+ features like dark mode, adaptive contrast, and motion consistency They align directly with the Play+ pillars: Distinct: Establishes a recognizable visual language Intuitive: Simplifies design handoff and dev workflow Inclusive: Ensures accessible contrast and adaptive states Play+ Token Architecture ■ Play+ uses a three-layered token model: 1. Global Tokens (Play+ defined) ■ Raw, context-free values such as colors, fonts, and spacing. Defined internally and optionally extendable by advanced teams. Example: --global-color-pink-500 : #e91e63 ; --global-font-family-body : "Inter" , sans-serif ; --global-radius-md : 0.75 rem ; 2. Semantic Tokens (User-defined) ■ Meaningful tokens that define roles — not values. These are defined in theme files, and map directly to either global tokens or raw values. Example: [data-theme = "custom"] { --color-brand-primary : var (--global-color-pink-500) ; --font-family-body : var (--global-font-family-body) ; --radius-md : var (--global-radius-md) ; } Play+ derives: Hover states Contrast-aware text colors Dark/light variants Disabled and focus styles 3. Component Tokens (Optional, scoped overrides) ■ Play+ components use semantic tokens by default. However, individual components (like buttons or modals) may expose their own token hooks for further flexibility. Use these only if you need to override global design decisions in specific components. The available tokens for each component are documented within that component's specific API guide. Example: --button-primary-background : var (--color-surface-interactive-default) ; --button-primary-text : var (--color-text-on-brand) ; --card-border-radius : var (--radius-md) ; Use these only if you need to override global design decisions in specific components. Summary ■ You never

write or maintain SCSS to theme. Every design value is a token — defined once, reused everywhere. Themes are swapped at runtime using data-theme attributes. Play+ takes care of motion, accessibility, and variant generation. Tokens are not just style variables — they're the programmable foundation of the Play+ design language. Let your brand breathe — one token at a time.

```
--global-color-pink-500: #e91e63;
--global-font-family-body: "Inter", sans-serif;
--global-radius-md: 0.75rem;


---

[data-theme="custom"] {
  --color-brand-primary: var(--global-color-pink-500);
  --font-family-body: var(--global-font-family-body);
  --radius-md: var(--global-radius-md);
}


---

--button-primary-background: var(--color-surface-interactive-default);
--button-primary-text: var(--color-text-on-brand);
--card-border-radius: var(--radius-md);
```